

## REMOTE WINDOW CONTROL SYSTEM

**Publication number:** JP6266666 (A)

**Publication date:** 1994-09-22

**Inventor(s):** SASAMATA YOSHINORI

**Applicant(s):** HITACHI LTD

**Classification:**

- international: **G06F13/00; G06F15/00; G09G5/14; G06F13/00; G06F15/00; G09G5/14; (IPC1-7): G06F15/00; G06F13/00; G09G5/14**

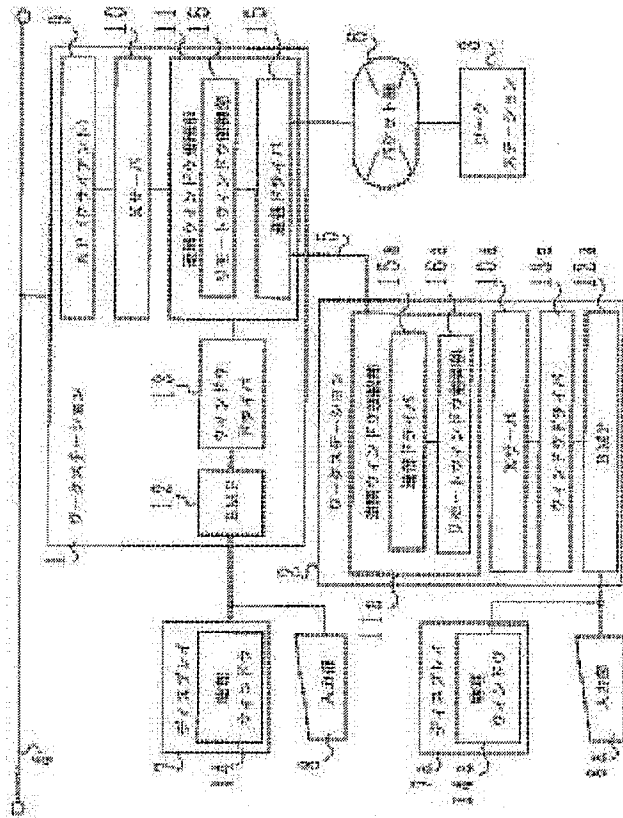
- European:

**Application number:** JP19930050379 19930311

**Priority number(s):** JP19930050379 19930311

### Abstract of JP 6266666 (A)

**PURPOSE:** To efficiently and inexpensively connect plural bases with the small number of terminals such as plural small-scaled LAN. **CONSTITUTION:** A system controls respective windows in UNIX devices which are provided with X window functions and are installed at remote places. Remote window connection parts 11 and 11a which connection-control the X servers 10 and 10a of the UNIX devices installed at the remote places convert command lines which the X servers 10 and 10a issue into simple protocols and control communication between the client of the other UNIX device-side and the X server of the self UNIX device through the X server of the other UNIX device being a connection destination are provided.



Data supplied from the esp@cenet database — Worldwide

特開平6-266666

(43)公開日 平成6年(1994)9月22日

(51)Int.Cl. <sup>5</sup>	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 15/00	3 1 0 R	7459-5L		
13/00	3 5 4 D	7368-5B		
G 0 9 G 5/14		8121-5G		

審査請求 未請求 請求項の数6 OL (全9頁)

(21)出願番号 特願平5-50379

(22)出願日 平成5年(1993)3月11日

(71)出願人 000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72)発明者 笹俣 吉典

神奈川県川崎市幸区鹿島田890番地の12

株式会社日立製作所情報システム事業部内

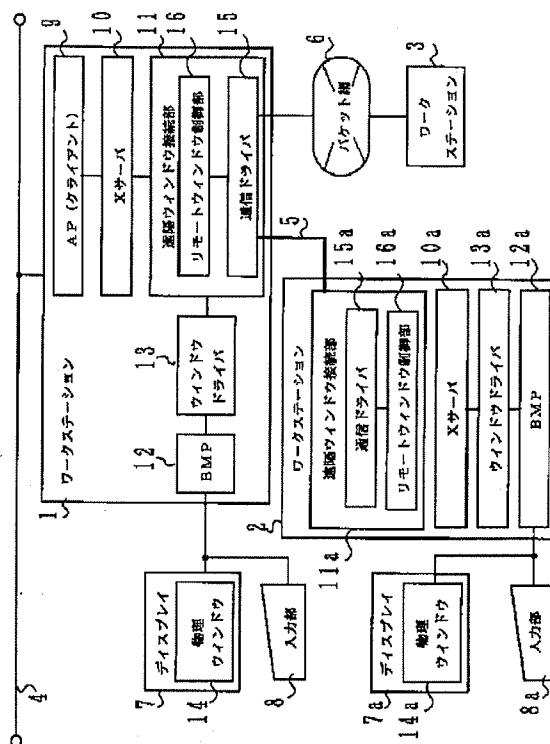
(74)代理人 弁理士 磯村 雅俊

(54)【発明の名称】 遠隔ウィンドウ制御システム

(57)【要約】

【目的】 複数の小規模LANの相互接続など、端末の数が少ない複数の拠点の接続を、効率良く、かつ、安価に行う。

【構成】 Xウィンドウ機能を有する各々遠隔地に設けられたUNIX装置間で、各々のウィンドウを制御するシステムであり、遠隔地に設けられた各々のUNIX装置のXサーバ間の接続制御を相互に行い、Xサーバが発行するコマンドラインを簡易なプロトコルに変換して、他UNIX装置側のクライアントと自UNIX装置のXサーバとの通信を、接続先他UNIX装置のXサーバを介して制御する遠隔ウィンドウ接続部を設ける構成とする。



## 【特許請求の範囲】

【請求項1】 アプリケーション処理を行うクライアントと、該クライアントとの通信による物理ウィンドウの表示制御を含む入出力処理を行うXサーバとからなるXウィンドウ機能を有する各々遠隔地に設けられたUNIX装置間で、各々のウィンドウを制御するシステムであり、上記遠隔地に設けられた各々のUNIX装置のXサーバ間の接続制御を相互に行い、接続先の他UNIX装置のXサーバを介して、該他UNIX装置側のクライアントと自UNIX装置のXサーバとの通信を制御する遠隔ウィンドウ接続手段を設けることを特徴とする遠隔ウィンドウ制御システム。

【請求項2】 請求項1に記載の遠隔ウィンドウ制御システムにおいて、上記遠隔ウィンドウ接続手段は、上記接続先の他UNIX装置間の回線に対応する通信制御を行なう通信制御手段と、該通信制御手段と上記自Xサーバ、および、自UNIX装置のウィンドウ入出力制御を行なうウィンドウ制御手段のそれぞれから入力される入力イベントの送信先を判別し、送信先に対応した上記入力イベントの送信処理を行なうリモートウィンドウ制御手段とを具備することを特徴とする遠隔ウィンドウ制御システム。

【請求項3】 請求項1、もしくは、請求項2のいずれかに記載の遠隔ウィンドウ制御システムにおいて、上記遠隔ウィンドウ接続手段は、上記接続先の他UNIX装置の物理ウィンドウに対して自Xサーバが発行するコマンドラインを、所定の圧縮手順により、簡易なプロトコルに変換する簡易プロトコル生成手段と、接続先の他UNIX装置の上記簡易プロトコル生成手段で簡易化されて送られてきた簡易なプロトコルを、上記所定の圧縮手順に対応する伸張手順により、該簡易なプロトコルの送信元の他UNIX装置のXサーバが発行したコマンドラインに変換するコマンドライン生成手段とを具備することを特徴とする遠隔ウィンドウ制御システム。

【請求項4】 請求項1から請求項3のいずれかに記載の遠隔ウィンドウ制御システムにおいて、上記遠隔ウィンドウ接続手段は、自UNIX装置側のクライアントからの描画リクエストに対応して自Xサーバが発行するコマンドラインに基づき、上記描画リクエストの送信先を判別し、判別した送信先に対応して、上記描画リクエストの送信処理を行なう第1のリモートウィンドウ制御手段を具備することを特徴とする遠隔ウィンドウ制御システム。

【請求項5】 請求項1から請求項4のいずれかに記載の遠隔ウィンドウ制御システムにおいて、上記遠隔ウィンドウ接続手段は、自UNIX装置の入力手段からの入力イベント内容に基づき、該入力イベントの送信先を判別し、判別した送信先に対応して、上記自入力手段からの入力イベントの送信処理を行なう第2のリモートウィンドウ制御手段を具備することを特徴とする遠隔ウィンドウ

制御システム。

【請求項6】 請求項1から請求項5のいずれかに記載の遠隔ウィンドウ制御システムにおいて、上記遠隔ウィンドウ接続手段は、上記接続先の他UNIX装置からの入力イベント内容に基づき、該入力イベントの送信先を判別し、判別した送信先に対応して、上記接続先の他UNIX装置からの入力イベントの送信処理を行なう第3のリモートウィンドウ制御手段を具備することを特徴とする遠隔ウィンドウ制御システム。

## 【発明の詳細な説明】

## 【0001】

【産業上の利用分野】本発明は、Xウィンドウ機能を有する各々遠隔地に設けられたUNIX装置間の接続を制御する技術に係り、特に、複数の小規模なLAN (Local Area Network) 間等の接続を、効率的、かつ、安価に行うのに好適な遠隔ウィンドウ制御システムに関するものである。

## 【0002】

【従来の技術】従来、複数の小規模LANを相互接続して一つのネットワーク環境にまとめ、例えば、データベースの地域毎の保守・運用を不要として、「管理の集中と、処理の分散」を図る技術がある。従来の技術でこのようなシステムを構築する場合、一般的には、LAN間を接続するものや、UNIX装置としてのワークステーション同士をパケット網で接続するものがある。尚、UNIX (ユニックス) は、AT&Tベル研究所が開発したOS (オペレーティングシステム) であり、時分割制御を行い、同時に複数ユーザ/複数タスクの実行をサポートする。

【0003】上述のLAN間を接続する技術では、いわゆるルータやリモートブリッジなどを用いて、LAN間を、専用線やパケット網、ISDN等で接続し、拠点のLANにUNIX装置 (または、X端末) を設置し、XウィンドウシステムのXプロトコルを用いて、遠隔地のディスプレイにウィンドウを表示する。または、それぞれのワークステーションにアプリケーションプログラム (以下、APと記載) を作成して、TCP/IP (Transmission Control Protocol/Internet Protocol) のプロトコルでAP間通信を行う。また、パケット網で接続する技術では、ワークステーション側にX.25プロトコルをサポートし、ワークステーション同士をパケット網で接続して、TCP/IPプロトコルでAP間通信を行う。

【0004】尚、Xウィンドウシステム (X Window System) は、米マサチューセッツ工科大学 (MIT) のAthenaプロジェクトで開発されたUNIX環境用マルチウィンドウ表示管理ソフトウェアであり、UNIX装置では、ほぼ必須のソフトウェアとして普及している。Xウィンドウシステム (以下、Xウィ

ンドウと記載する)は、階層構造をなす短形のウィンドウを複数オーバーラップ表示し、各ウィンドウに文字や図形、画像を表示する基本機能を備えている。いわゆるXサーバが、画面制御や、キーボード/マウス入力などを一括して管理する。ユーザプログラムは、クライアントと呼ばれ、UNIXのプロセス通信機能を使い、Xサーバに処理を依頼する。この構成を、サーバー・クライアント方式と呼ぶ。このユーザプログラムへのプログラミング・インタフェースを提供するライブラリがXlibである。Xウィンドウ(Xサーバ)の機能だけを専用ハードウェア化した「X端末」も開発されている。

【0005】Xウィンドウでのウィンドウ操作の流れは、クライアントであるAPがXlibの関数を用いて発行した描画のリクエストを、Xサーバが解釈し、ディスプレイ表示やキーボード入力等のウィンドウ入出力制御を行なうウィンドウ制御部(以下、ウィンドウドライバと記載)に対して、複数のコマンドラインを発行する。そして、このウィンドウドライバが、ビットマッププロセッサ(以下、BMP:Bit Map Processorと記載)を用いて、ディスプレイに表示する。キーボードやマウスからの入力は、BMPがそれを感じ、ウィンドウドライバに割込みを発生させる。ウィンドウドライバは、キーボードやマウスからの入力内容をXサーバに通知し、Xサーバは、ウィンドウを表示しているAPに、その内容をイベントとして通知する。尚、このようなXウィンドウに関しては、「スーパーアスキー 1991年2月号(「SuperASCII, vol. 2 #12 December 1991」)(1991年(株)アスキー発行)の第73頁から第88頁に記載されている。

【0006】しかし、このような従来技術による複数の小規模LANの相互接続では、以下のような問題があった。まず、LAN間の接続技術では、ルータやリモートブリッジが高価であり、多拠点に設置するとコストがかかる。また、LAN上の高速(例えば、10Mbps(メガビット/秒))な転送速度と比べて、専用線やパケット網等は低速であり、Xウィンドウの冗長性のあるXプロトコルでは性能面で問題があり、LAN間接続でのX端末の使用は業務的に不可能である。このような性能面を解決するためには、アプリケーションで通信のオーバーヘッドを考慮する必要がある。また、AP間通信で接続を行うと、各拠点のワークステーションに業務アプリケーションを配置する必要があり、沢山ある遠隔地の各拠点のワークステーションのメンテナンスの方式を検討する必要があった。また、パケット網接続方式でも、LAN間接続方式と同様に、性能とメンテナンスに問題がある。

【0007】

【発明が解決しようとする課題】解決しようとする問題は、従来の技術では、高価なルータやリモートブリッ

ジの使用や、アプリケーションでの通信のオーバーヘッドの考慮、各拠点のワークステーションへの業務アプリケーションの配置、および、遠隔地の多拠点のワークステーションのメンテナンス方式の検討等が必要であり、複数の小規模LANの相互接続を、効率良く、かつ、安価に行うことができない点である。本発明の目的は、これら従来技術の課題を解決し、複数の小規模LANの相互接続などを、安価な構成で効率良く行うことを可能とする遠隔ウィンドウ制御システムを提供することである。

【0008】

【課題を解決するための手段】上記目的を達成するため、本発明の遠隔ウィンドウ制御システムは、(1)アプリケーション処理を行うクライアントと、このクライアントとの通信による物理ウィンドウの表示制御を含む入出力処理を行うXサーバとからなるXウィンドウ機能を有する各々遠隔地に設けられたUNIX装置間で、各々のウィンドウを制御するシステムであり、遠隔地に設けられた各々のUNIX装置のXサーバ間の接続制御を相互に行い、接続先の他UNIX装置のXサーバを介して、この他UNIX装置側のクライアントと自UNIX装置のXサーバとの通信を制御する遠隔ウィンドウ接続部を設けることを特徴とする。また、(2)上記(1)に記載の遠隔ウィンドウ制御システムにおいて、遠隔ウィンドウ接続部は、接続先の他UNIX装置間の回線に対応する通信制御を行なう通信制御部と、この通信制御部と自Xサーバ、および、自UNIX装置のウィンドウ入出力制御を行なうウィンドウ制御部のそれぞれから入力される入力イベントの送信先を判別し、送信先に対応した入力イベントの送信処理を行なうリモートウィンドウ制御部とを具備することを特徴とする。また、(3)上記(1)、もしくは、(2)のいずれかに記載の遠隔ウィンドウ制御システムにおいて、遠隔ウィンドウ接続部は、接続先の他UNIX装置の物理ウィンドウに対して自Xサーバが発行するコマンドラインを、所定の圧縮手順により、簡易なプロトコルに変換する簡易プロトコル生成部と、接続先の他UNIX装置の簡易プロトコル生成部で簡易化されて送られてきた簡易なプロトコルを、上述の所定の圧縮手順に対応する伸張手順により、この簡易なプロトコルの送信元の他UNIX装置のXサーバが発行したコマンドラインに変換するコマンドライン生成部とを具備することを特徴とする。また、(4)上記(1)から(3)のいずれかに記載の遠隔ウィンドウ制御システムにおいて、遠隔ウィンドウ接続部は、自UNIX装置側のクライアントからの描画リクエストに対応して自Xサーバが発行するコマンドラインに基づき、描画リクエストの送信先を判別し、判別した送信先に対応して、描画リクエストの送信処理を行なう第1のリモートウィンドウ制御部を具備することを特徴とする。また、(5)上記(1)から(4)のいずれかに記載の遠隔ウィンドウ制御システムにおいて、遠隔

ウィンドウ接続部は、自UNIX装置の入力部からの入力イベント内容に基づき、この入力イベントの送信先を判別し、判別した送信先に対応して、自入力部からの入力イベントの送信処理を行なう第2のリモートウィンドウ制御部を具備することを特徴とする。また、(6)上記(1)から(5)のいずれかに記載の遠隔ウィンドウ制御システムにおいて、遠隔ウィンドウ接続部は、接続先の他UNIX装置からの入力イベント内容に基づき、この入力イベントの送信先を判別し、判別した送信先に対応して、接続先の他UNIX装置からの入力イベントの送信処理を行なう第3のリモートウィンドウ制御部を具備することを特徴とする。

【0009】

【作用】本発明においては、Xウィンドウ機能を有するUNIX装置は、相互のXサーバを介して接続されているので、例えば、異なるLANのクライアントとの通信を、ルータやリモートブリッジを必要としない安価な構成で行うことができる。また、相互のXサーバ間の通信に、Xサーバが発行する複数のコマンドラインを、例えば、1命令にするプロトコルを用いることも容易にでき、通信のオーバーヘッドを軽減できる。

【0010】

【実施例】以下、本発明の実施例を、図面により詳細に説明する。図1は、本発明の遠隔ウィンドウ制御システムの本発明に係わる構成の一実施例を示すブロック図である。本図において、1〜3は、Xウィンドウ機能を有する各々遠隔地に設けられたUNIX装置としてのワークステーション、4は、ワークステーション1を収容するLAN（図中、イーサネットと記載）、5は、ワークステーション1、2間を接続する専用線、6は、ワークステーション3を収容するパケット網、7および7aは、それぞれ、ワークステーション1およびワークステーション2に接続されたCRT（Cathode Ray Tube、陰極線管）等のディスプレイ、8および8aは、それぞれ、ワークステーション1およびワークステーション2に接続されたキーボードやマウス等からなる入力部、9は、ワークステーション1に収容されているアプリケーション処理を行うクライアント、10および10aは、クライアントとの通信によるウィンドウの表示制御を含む入出力処理を行うXサーバ、11および11aは、本発明に係わり、ワークステーション2のXサーバ10aとワークステーション1のクライアント9との接続制御を行なう遠隔ウィンドウ接続部、12、12a、および、13、13aは、それぞれ、ワークステーション1、2に接続されたディスプレイ5、5aへの物理ウィンドウ14、14aの描画や、入力部8、8aからの入力等の入出力制御を行なうBMPとウィンドウドライバ（ウィンドウ制御部）、15および15aは、それぞれ、遠隔ウィンドウ接続部11、11aに設けられ、接続先のワークステーション1、2間の回線

（専用線5）に対応する通信制御を行なう本発明の通信制御部としての通信ドライバ、16および16aは、それぞれ、遠隔ウィンドウ接続部11、11aに設けられ、通信ドライバ15、15aやXサーバ10、10a、または、ウィンドウドライバ13、13aのそれぞれから入力される入力イベントの送信先を判別し、送信先に対応した入力イベントの送信処理を行なうリモートウィンドウ制御部である。尚、ワークステーション3においても、ワークステーション1、2と同様に、本発明に係わる遠隔ウィンドウ接続部が設けられており、パケット網6を介して、ワークステーション1に接続され、クライアント9の利用が可能な構成となっている。

【0011】このような構成の遠隔ウィンドウ制御システムにおいて、例えば、ワークステーション1のクライアント9が、Xlibの関数の描画リクエストを発行すると、Xサーバ10が、このリクエスト内容を解釈して、ウィンドウドライバ13に対して複数のコマンドラインを発行するが、本実施例の遠隔ウィンドウ制御システムでは、そのコマンドを、遠隔ウィンドウ接続部11を経由するようにする。そして、遠隔ウィンドウ接続部11は、リモートウィンドウ制御部16により、Xサーバ10からのコマンドラインが、どの物理ウィンドウに対してのリクエストかを判断し、該当する物理ウィンドウに送信処理を行なう。例えば、ワークステーション1の物理ウィンドウ14に対してのリクエストの場合、Xサーバ10からのコマンドラインを、そのままウィンドウドライバ13に発行する。また、遠隔地にあるワークステーション2の物理ウィンドウ14aに対してのリクエストの場合には、Xサーバ10からの複数のコマンドラインを解析して、通信ドライバ15を用いて、目的の物理ウィンドウ14aの存在するワークステーション2と通信を行う。この時、リモートウィンドウ制御部16は、コマンドラインを、所定の圧縮手順により、簡易なプロトコルに変換すると共に、専用線5やパケット網6での通信を可能にするためのハンドリングを通信ドライバ15に行なう。

【0012】ワークステーション2においては、遠隔ウィンドウ接続部11aが、通信ドライバ15aにより、ワークステーション1の遠隔ウィンドウ接続部11からのリクエストを受信し、そして、リモートウィンドウ制御部16aにより、このリクエストから、ワークステーション1のXサーバ10と同じコマンドラインを生成して発行する。その後、ウィンドウドライバ13aが、BMP12aを用いてディスプレイ7aの物理ウィンドウ14aに表示する。また、ワークステーション2の入力部8aから、ワークステーション1のクライアント9への入力、遠隔ウィンドウ接続部11aが、BMP12aとウィンドウドライバ13aにより、このリクエスト（入力イベント）を受信し、リモートウィンドウ制御部16により、送信先のワークステーション1を判別し、

そして、通信ドライバ15aを用いて、ワークステーション1の遠隔ウィンドウ制御部11に送信する。ワークステーション1の遠隔ウィンドウ接続部11は、通信ドライバ15で受信したワークステーション2からのリクエスト（入カイベント）先のクライアント9を、リモートウィンドウ制御部16で判別し、Xサーバ10に送信する。尚、ワークステーション1において、自装置に接続されている入力部8からクライアント9への入力があれば、遠隔ウィンドウ接続部11が、ウィンドウドライバ13からのイベントをそのままXサーバ10に送信する。

【0013】このような動作を行なう遠隔ウィンドウ接続部11の詳細を、次の図2を用いて説明する。図2は、図1における遠隔ウィンドウ接続部の本発明に係わる構成の一実施例を示すブロック図である。本実施例の遠隔ウィンドウ接続部11は、自ワークステーション側のクライアント9からの描画リクエストに対応して自Xサーバ10が発行するコマンドラインに基づき、描画リクエストの送信先を判別し、判別した送信先に対応して、描画リクエストの送信処理を行なう第1のリモートウィンドウ制御部21と、自ワークステーションのウィンドウドライバ13からの入カイベント内容を解析して、この入カイベントの送信先を判別し、判別した送信先に対応して、自ワークステーションのウィンドウドライバ13からの入カイベントの送信処理を行なう第2のリモートウィンドウ制御部22と、通信ドライバ15を介して入力される接続先の他ワークステーションからの入カイベント内容を解析して、この入カイベントの送信先を判別し、判別した送信先に対応して、接続先の他ワークステーションからの入カイベントの送信処理を行なう第3のリモートウィンドウ制御部23と、第1のリモートウィンドウ制御部21と第3のリモートウィンドウ制御部23が、それぞれの送信処理時に参照するプロトコル変換テーブル24とにより構成されている。

【0014】また、第1のリモートウィンドウ制御部21には、接続先の他ワークステーションの物理ウィンドウに対して自Xサーバが発行するコマンドラインを、プロトコル変換テーブル24を参照して、簡易なプロトコルに変換する簡易プロトコル生成部25が、また、第3のリモートウィンドウ制御部23には、接続先の他ワークステーションの簡易プロトコル生成部で簡易化されて送られてきた簡易なプロトコルを、プロトコル変換テーブル24を参照して、この簡易なプロトコルの送信元の他ワークステーションのXサーバが発行したコマンドラインに変換するコマンドライン生成部26がそれぞれ設けられている。

【0015】第2のリモートウィンドウ制御部22は、ウィンドウドライバ13から発生するイベント及び、割込みを認識し、このイベントの内容を解析する。そして、このイベントが、自ワークステーション側のクライ

アント9に対するものであれば、ウィンドウドライバ13からのイベントをXサーバ10に送信する。また、このイベントが、遠隔地で接続されているワークステーション側のクライアントに対するものであれば、このウィンドウドライバ13からのイベント内容を、通信ドライバ15を用いて遠隔地のワークステーションに送信する。

【0016】第1のリモートウィンドウ制御部21は、Xサーバ10から発生するイベント及び、割込みを認識し、このイベント、すなわち、描画リクエストに対応してXサーバ10からBMPに対して発行される複数のコマンドラインを解析する。そして、このコマンドラインが、自ワークステーションの物理ウィンドウに対するものであれば、このコマンドラインをウィンドウドライバ13に送信し、自ワークステーションのディスプレイに表示させる。また、このコマンドラインが、遠隔地で接続されているワークステーションの物理ウィンドウに対するものであれば、このXサーバ10からのコマンドラインを、通信ドライバ15を用いて遠隔地のワークステーションに送信する。

【0017】この時、第1のリモートウィンドウ制御部21は、Xサーバ10から発行された複数のコマンドがXlibのどの関数が認識し、簡易プロトコル生成部25により、この認識した関数をもとに、通信ドライバ15に送信するデータを、通信オーバーヘッドの小さいプロトコルに編集する。すなわち、簡易プロトコル生成部25は、プロトコル変換テーブル24を参照して、この関数が示す処理を簡易なプロトコルに変換する。プロトコル変換テーブル24には、Xサーバ10が発行するコマンドが示す所定の一連の処理、例えば、線を描画する処理など、各一連の処理ごとの識別番号が登録されており、簡易プロトコル生成部25は、Xサーバ10が発行するコマンドラインを、対応する処理の識別番号に変換する。例えば、線を描画する処理であれば、識別番号「001」に変換する。そして、第1のリモートウィンドウ制御部21は、この識別番号「001」と線の色や始点、終点などのパラメータを、通信ドライバ15を用いて遠隔地のワークステーションに送信する。Xサーバ10が発行するコマンドラインのプロトコルは、一般に、冗長であり、低速な転送速度の専用線やパケット網で、転送を行なうのは無理であるが、このように、コマンドラインを、予め設定されている対応する処理の識別番号とパラメータに変換して簡易化することにより、Xサーバ10が発行するコマンドラインの内容を、専用線やパケット網を介して転送することができる。

【0018】第3のリモートウィンドウ制御部23は、通信ドライバ15から発生するイベント及び、割込みを認識し、このイベントを解析する。そして、このイベントが、自ワークステーションのクライアントに対するものであれば、このイベントをサーバ10に送信する。ま

た、このイベントが、自ワークステーションの物理ウィンドウに対するものであれば、すなわち、他ワークステーション側のクライアントからの描画リクエストであれば、このイベント内容をウィンドウドライバ13に送信し、自ワークステーションのディスプレイに表示させる。

【0019】ここで、この描画リクエストであるイベントは、他ワークステーション側で、プロトコル変換テーブル24と同じテーブルを用いて簡易化されたプロトコルである。そのため、第3のリモートウィンドウ制御部23は、コマンドライン生成部26により、プロトコル変換テーブル24を用いて、簡易化されたプロトコルから、他ワークステーションのXサーバが発行したコマンドラインを再生成する。すなわち、コマンドライン生成部26は、プロトコル変換テーブル24を参照して、このイベントが示す識別番号に対応する一連の処理を選出し、この処理に対応する他ワークステーション側のXサーバがクライアントの描画リクエストに対応して発行したコマンドラインを発行する。

【0020】次に、遠隔ウィンドウ接続部11の本発明に係わる動作説明を、次の図3を用いて行なう。図3は、図2における遠隔ウィンドウ接続部の本発明に係わる処理動作の一実施例を示すフローチャートである。図2におけるXサーバ10からイベントが入力されたならば(ステップ301、302)、図2の第1のリモートウィンドウ制御部21は、この入力イベントの内容を解析して、その送信先を判別する(ステップ303)。この入力イベントが、自装置の物理ユニットの描画リクエストであれば(ステップ304)、入力イベント(描画リクエストに対応して図2のXサーバ10が発行したコマンドライン)を、自装置のウィンドウドライバに送信する(ステップ305)。また、他装置の物理ユニットの描画リクエストであれば(ステップ304)、図2の簡易プロトコル生成部25により、このコマンドラインを簡易なプロトコルに変換して(ステップ306)、図2の通信ドライバ13を介して、送信先の他装置に送信する(ステップ307)。

【0021】また、入力イベントが、自装置のウィンドウドライバから入力されたならば(ステップ302)、図2の第2のリモートウィンドウ制御部22は、この入力イベントの内容を解析して、その送信先を判別する(ステップ308)。この入力イベントが、自装置側のクライアントへの入力であれば(ステップ309)、この入力イベント(図1の入力部8からの入力データ)を、自装置のXサーバに送信する(ステップ310)。また、他装置側のクライアントへの入力であれば(ステップ309)、図2の通信ドライバ13を介して、送信先の他装置に送信する(ステップ311)。

【0022】また、入力イベントが、通信ドライバから入力されたならば(ステップ302)、図2の第3のリ

モートウィンドウ制御部23は、この入力イベントの内容を解析して、その送信先を判別する(ステップ312)。この入力イベントが、自装置の物理ユニットへの描画リクエスト、すなわち、簡易化されたプロトコルであれば(ステップ313)、図2のコマンドライン生成部26により、この簡易なプロトコルを元のコマンドラインに変換して(ステップ314)、自装置のウィンドウドライバに送信する(ステップ315)。また、自装置側のクライアントに対する入力リクエストであれば(ステップ313)、この入力リクエストを、自装置のXサーバへ送信する(ステップ316)。

【0023】以上、図1～図3を用いて説明したように、本実施例の遠隔ウィンドウ制御システムでは、それぞれ遠隔地に設けられたXウィンドウ機能を有するワークステーションを、相互のXサーバを介して接続しているので、例えば、異なるLANのクライアントとの通信を、ルータやリモートブリッジを必要としない安価な構成で、従来のクライアント(AP)とXサーバのインターフェースを保証して行うことができる。また、相互のXサーバ間の通信に、Xサーバが発行する複数のコマンドラインを、例えば、1命令にするプロトコルを用いており、通信のオーバーヘッドを軽減でき、アプリケーションで通信のオーバーヘッドを考慮する必要がなく、システムを高性能化できる。また、従来のAP間通信で接続を行う場合のように、各拠点のワークステーションに業務アプリケーションを配置する必要がなくなり、遠隔地の各拠点のワークステーションのメンテナンスの検討が不要となり、システムの運用が容易となる。

【0024】尚、本発明は、図1～図3を用いて説明した実施例に限定されるものではなく、その要旨を逸脱しない範囲において種々変更可能である。例えば、図1におけるワークステーション2を、Xサーバ専用機とすることにより、ワークステーション2の遠隔ウィンドウ接続部11aにおける第2、第3のリモートウィンドウ制御部のそれぞれを不要とし、その構成を簡素化することも可能である。

#### 【0025】

【発明の効果】本発明によれば、高価なルータやリモートブリッジの使用や、アプリケーションでの通信のオーバーヘッドの考慮、各拠点のUNIX装置としてのワークステーションへの業務アプリケーションの配置、および、遠隔地の多拠点のワークステーションのメンテナンス方式の検討等が不要となり、複数の小規模LAN等、端末の数が少ない複数の拠点の相互接続を、効率良く、かつ、安価に行うことが可能である。

#### 【図面の簡単な説明】

【図1】本発明の遠隔ウィンドウ制御システムの本発明に係わる構成の一実施例を示すブロック図である。

【図2】図1における遠隔ウィンドウ接続部の本発明に係わる構成の一実施例を示すブロック図である。

11

12

【図3】図2における遠隔ウィンドウ接続部の本発明に係わる処理動作の一実施例を示すフローチャートである。

【符号の説明】

1～3 ワークステーション

4 LAN

5 専用線

6 パケット網

7、7a ディスプレイ

8、8a 入力部

9 クライアント

10、10a Xサーバ

11、11a 遠隔ウィンドウ接続部

12、12a BMP

13、13a ウィンドウドライバ

14、14a 物理ウィンドウ

15、15a 通信ドライバ

16、16a リモートウィンドウ制御部

21 第1のリモートウィンドウ制御部

22 第2のリモートウィンドウ制御部

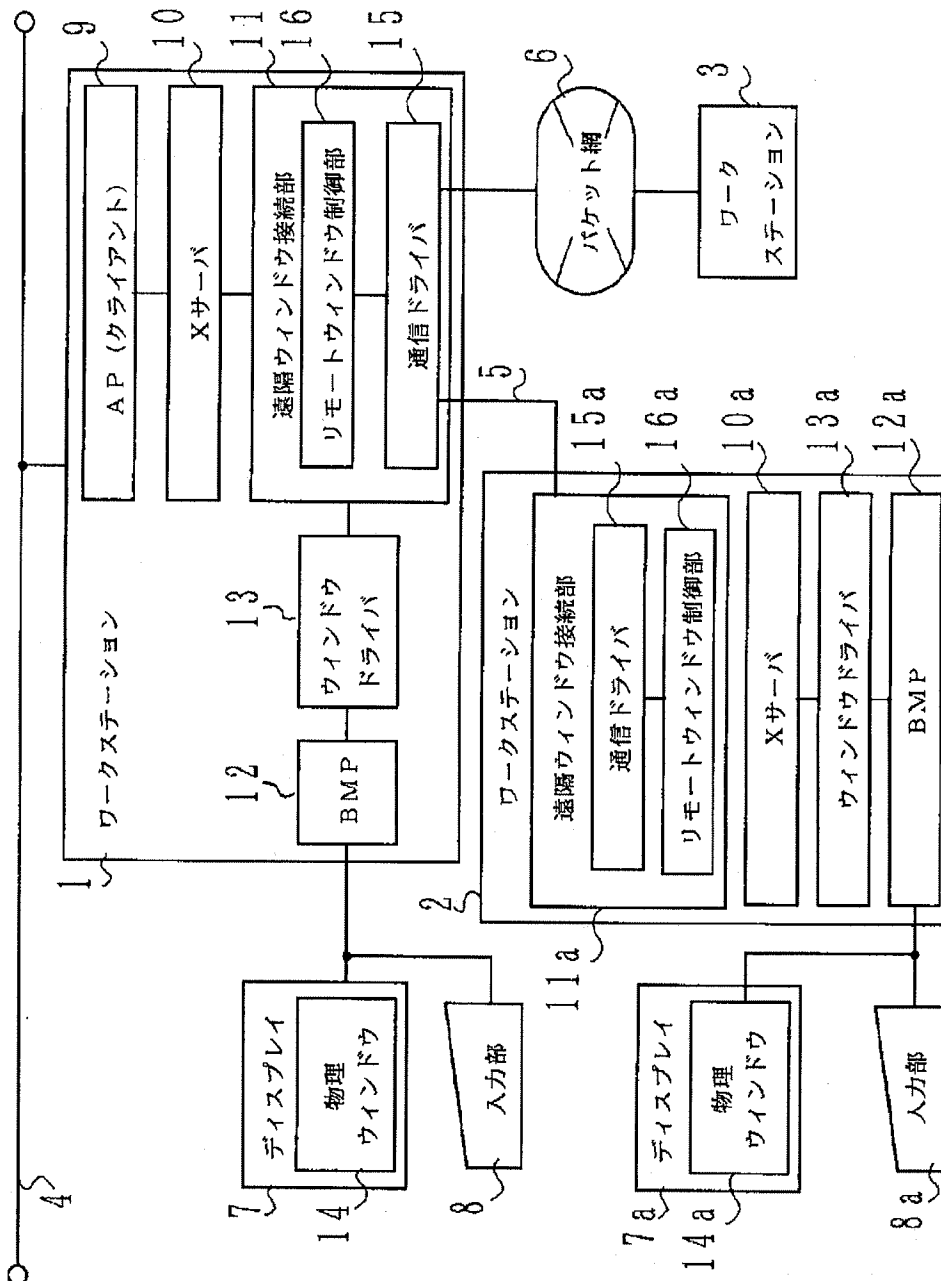
23 第3のリモートウィンドウ制御部

10 24 プロトコル変換テーブル

25 簡易プロトコル生成部

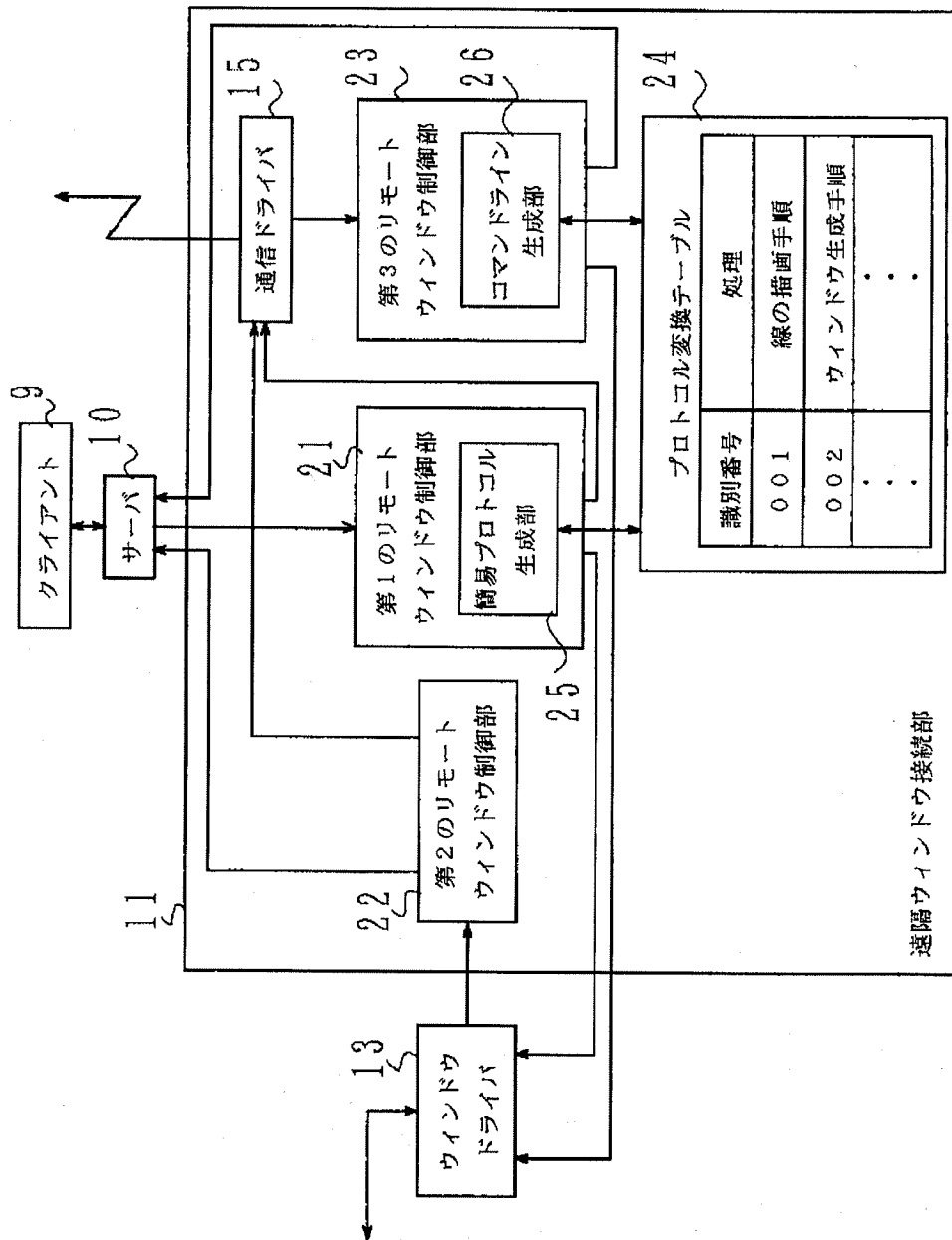
26 コマンドライン生成部

【図1】





【図2】



【図3】

